This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (previously presented): A method for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said method comprising:

specifying a set of constraints on said backup tunnels;

specifying an optimality function; and

performing linear programming operations based on said set of constraints and said optimality function to find said backup tunnels, wherein performing linear programming operations based on said set of constraints and said optimality function includes minimizing said optimality function while satisfying said set of constraints and wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

Claim 2 (original): The method of claim 1 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

Claim 3 (original): The method of claim 2 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

Claim 4 (previously presented): The method of claim 3 wherein said optimality function includes a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

Claim 5 (original): The method of claim 1 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

Claim 6 (original): The method of claim 1 wherein said linear programming operations are performed on a network that does not include a protected link.

Claim 7 (original): The method of claim 1 wherein performing linear programming operations comprises:

finding links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

identifying backup tunnels made up of said links.

Claim 8 (original): The method of claim 5 wherein identifying backup tunnels made up of said links comprises:

establishing a subnetwork consisting of only links found by said linear programming procedure;

identifying a shortest path between said first node and said second node through said subnetwork;

identifying a link in said shortest path having a smallest remaining bandwidth;

eliminating said link having said smallest remaining bandwidth from said subnetwork;

reducing available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

repeating establishing, identifying said shortest path, identifying said link in said shortest path, eliminating, and reducing until no links remain in said subnetwork.

Claim 9 (previously presented): A computer-readable medium encoded for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said computer-readable medium comprising:

code that specifies a set of constraints on said backup tunnels;

code that specifies and optimality function; and

code that performs linear programming operations based on said set of constraints and said optimality function to find said backup tunnels, wherein said code that performs linear programming operations based on said set of constraints and said optimality function includes code that minimizes said optimality function while satisfying said set of constraints and wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

Claim 10 (previously presented): The computer-readable medium claim 9 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

Claim 11 (previously presented): The computer-readable medium of claim 10 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

Claim 12 (previously presented): The computer-readable medium of claim 11 wherein said optimality function includes a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

Claim 13 (previously presented): The computer-readable medium of claim 9 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

Claim 14 (previously presented): The computer-readable medium of claim 9 wherein said linear programming operations are performed on a network that does not include a protected link.

Claim 15 (previously presented): The computer-readable medium of claim 9 wherein said code that performs linear programming operations comprises:

code that finds links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

code that identifies backup tunnels made up of said links.

Claim 16 (previously presented): The computer-readable medium of claim 15 wherein said code that identifies backup tunnels made up of said links comprises:

code that establishes a subnetwork consisting of only links found by said linear programming procedure;

code that identifies a shortest path between said first node and said second node through said subnetwork;

code that identifies a link in said shortest path having a smallest remaining bandwidth;

code that eliminates said link having said smallest remaining bandwidth from said subnetwork;

code that reduces available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

code that repeatedly invokes said code that identifies said shortest path, said code that identifies said link in said shortest path, said code that eliminates, and said code that reduces until no links remain in said subnetwork.

Claim 17 (previously presented): Apparatus for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said apparatus comprising:

a processor; and

a memory device storing instructions executed by said processor, said instructions comprising:

code that specifies a set of constraints on said backup tunnels;

code that specifies an optimality function; and

code that performs linear programming operations based on said set of constraints and said optimality function to find said backup tunnels, wherein said code that performs linear programming operations based on said set of constraints and said optimality function includes code that minimizes said optimality function while satisfying said set of constraints and wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

Claim 18 (original): The apparatus of claim 17 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

Claim 19 (original): The apparatus of claim 18 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and

for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

Claim 20 (previously presented): The apparatus of claim 18 wherein said optimality function includes a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

Claim 21 (original): The apparatus of claim 17 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

Claim 22 (original): The apparatus of claim 17 wherein said linear programming operations are performed on a network that does not include a protected link.

Claim 23 (original): The apparatus of claim 17 wherein said code that performs linear programming operations comprises:

code that finds links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

code that identifies backup tunnels made up of said links.

Claim 24 (original): The apparatus of claim 17 wherein said code that identifies backup tunnels made up of said links comprises:

code that establishes a subnetwork consisting of only links found by said linear programming procedure;

code that identifies a shortest path between said first node and said second node through said subnetwork;

code that identifies a link in said shortest path having a smallest remaining bandwidth;

code that eliminates said link having said smallest remaining bandwidth from said subnetwork;

code that reduces available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

code that repeatedly invokes said code that identifies said shortest path, said code that identifies said link in said shortest path, said code that eliminates, and said code that reduces until no links remain in said subnetwork.

Claim 25 (previously presented): Apparatus for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said apparatus comprising:

means for specifying a set of constraints on said backup tunnels;

means for specifying an optimality function; and

means for performing linear programming operations based on said set of constraints and said optimality function to find said backup tunnels, wherein said means for performing linear programming operations based on said set of constraints and said optimality function includes means for minimizing said optimality function while satisfying said set of constraints and wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

Claim 26 (previously presented): The method of claim 1 wherein said linear programming operations are performed on a network that includes a protected node and links connected to said protected node.